Text-CRNN+Attention 架构下的多类别文本信息分类 *

卢 健+，马成贤，杨腾飞，周嫣然

(西安工程大学 电子信息学院, 西安 710048)

摘 要：文本分类作为数据挖掘和信息检索领域的研究热点。迄今为止，传统机器学习方法依赖人工提取特征，复杂度高；深度学习网络本身特征表达能力强，但模型可解释性弱导致关键特征信息丢失。为此，以网络层次结合的方式设计了 CRNN 网络，并引入 Attention 机制，提出一种 Text-CRNN+Attention 模型。首先利用 CNN 处理局部特征的位置不变性，提取高效局部特征信息；然后 RNN 进行序列特征建模时，引入 Attention 机制对每一时刻输出序列信息进行自动加权，减少关键特征的丢失；最后完成时间和空间上的特征提取。实验结果表明，提出的模型较其他模型准确率提升了 2~3 个百分点；在提取文本特征时，该模型既保证了数据的局部相关性，又起到强化序列特征的有效组合能力。

关键词：文本分类；CNN；RNN；CRNN；Attention 机制

中图分类号：TP391.1 doi: 10.19734/j.issn.1001-3695.2018.12.0858

Multi-category text information classification with Text-CRNN+Attention architecture

Lu Jian+, Ma Chengxian, Yang Tengfei, Zhou Yanran

(*School of Electronic & Information*, *Xi`an Polytechnic University*, *Xi'an* 710048, *China*)

Abstract: Text classification is a research hotspot in the field of data mining and information retrieval. In view of the current research process , traditional machine learning methods relies on manual feature extraction with high complexity; Deep learning network has strong feature expression ability, but the model is weak in interpretability, leading to the loss of key feature information. For this reason, the author designed the CRNN network in the way of network level combination, introduced Attention mechanism, and proposed a Text-CRNN+Attention model. Firstly, CNN was used to deal with the position invariance of local features and extracted efficient local feature information. Then, Attention mechanism was introduced to automatically weigh the output sequence information at each time to reduce the loss of key features when RNN was used to model the sequence features. The feature extraction in time and space is completed. The experimental results show that the accuracy of the proposed model is 2 to 3 percentage points higher than that of other models. When dealing with text data, the model not only guarantees the local correlation of data, but also strengthens the effective combination ability of sequence features.

Key words: text classification; CNN; RNN; CRNN; Attention mechanism

0 引言

文本分类研究作为自然语言处理任务研究中不可获取的子任务，由于文本本身存在语料通用性差、评价标准不一等问题，其研究具备高度复杂性。

机器学习方法的应用是将文本分类任务拆分为特征工程和分类器，完成数据到信息、信息到知识的过程。机器学习新方法研究主要基于决策树、KNN、SVM、朴素贝叶斯分类、神经网络等[1,2]。例如，Goudjil 等人[3]采用 SVM 分类器的后验概率来选择样本进行分类，缓和了文本特征信息间关系；改进的树增强朴素贝叶斯(TAN)[4]有效提升了准确率，但计算过程复杂、耗时；Facebook[5]提出的 FastText 模型考虑了单词间的顺序关系和分类器的计算速度，模型结构简单、速度快，但准确率较低。虽然机器学习方法应用效果不断优化，但是过于依赖人工提取特征，导致特征表达能力弱、成本高。

由于图像、自然语言等数据本身具有连续性和局部相关性，深度学习技术作为端对端的解决方式，可以更好地表达文本特征信息。深度学习技术在文本分类中应用前提是解决词的分布式表示（词映射）问题，简单 One-Hot Representation 处理方式到 Mikolov 等人提出的关于词向量计算的 word2vec 模型[6,7]，促进了深度学习技术的应用。例如，Kim 等人[8]提出的 Text-CNN 模型，将词映射引入简易 CNN 模型，利用 CNN 捕捉数据的局部相关性来挖掘文本数据局部信息的重要性，但由于卷积核的固定，无法获取更长序列信息；Kalchbrenner 等人[9]利用宽卷积（wide convolution）代替窄卷积，并采用 K-max pooling 方式，一是解决了输入序列长度受限问题，二是减少了特征信息的丢失；CNN 虽然可以高效挖掘数据局部特征信息，但无法获取上下文信息。RNN 作为自然语言处理中的标配模型，可以更好地处理文本上下文信息，Sutskever 等人[10]利用隐藏层的循环递归来改进传统前向神经网络，以循环递归特性来挖掘数据序列信息；文献[11]中提出的双向 LSTM 结构，通过获取更长且双向的序列信息，

收稿日期：2018-12-01；**修回日期：**2019-02-26 **基金项目：**国家自然科学基金资助项目(51607133)；陕西省教育厅专项科学研究计划项目(17JK0332)；陕西省科技厅科技发展计划项目（2011K06-01）；西安市碑林区应用技术研发项目（GX1807）

作者简介：卢健（1978-），男（通信作者），河北人，副教授，硕导，博士，主要研究方向为 AI 机器人应用创新（406170365@qq.com）；马成贤（1996-），男（回族），青海民和人，硕士研究生，主要研究方向为深度学习、计算机视觉；杨腾飞（1996-），男，陕西兴平人，硕士研究生，主要研究方向为深度学习；周嫣然（1994-），女，陕西西安人，硕士研究生，主要研究方向为深度学习.

更好地表达了上下文信息；但由于 RNN 自身循环递归特性，网络结构复杂度高，处理数据比较耗时。CNN 和 RNN 处理文本数据信息时，虽然以自身结构特性提取不同特征来描述文本数据信息，但两者存在不足:可解释性弱。为此，Bahdanau 等人[12]将 Attention 机制引入 RNN，并在机器翻译任务研究中取得不俗的效果，开启了 Attention 机制在 NLP 领域中应用先河。Attention 机制的引入强化了 RNN 模型对序列信息高效挖掘能力，被广泛应用于语音识别、文本分析、会话模型等任务研究。由于 Attention 机制计算过程是可微的，文献[13]中将 Attention 机制用于 CNN 网络的卷积层、池化层和整个网络结构来验证 Attention 机制的普适性。Attention 机制的引入强化了网络挖掘文本特征信息的能力[14]，但在不同任务研究中，CNN、RNN 以自身网络结构展现了各自优势与不足，中国科学院研究人员[15] 结合 CNN 和 RNN 网络结构优势，提出了 TextRCNN 模型，虽然准确率只提升了 1%，但值得借鉴的是模型结构设计新思路；文献[16]中提出的 CRNN 模型，实现了端对端的特征学习方式，由局部特征信息到序列信息的组合过程，可以更好地描述数据特征信息。

不断改进的网络结构可以更好地描述数据特征信息，Attention 机制可以强化网络学习能力。因此，为了更好地解决文本数据处理过程中局部信息的相关性、序列信息的连贯性、可解释性弱等问题，在遵从网络结构设计通用性原则[17]下，通过网络层次结合的方式设计 CRNN 网络结构，并引入 Attention 机制，本文提出了一种 Text-CRNN+Attention 模型。模型汲取了 CNN、RNN、Attention 机制的优势，以循序渐进的方式，由局部特征提取到全局特征，再到序列信息的有效挖掘，同时体现了每一时刻输出序列信息的不同权重，起到增强网络学习能力的效果，实现文本数据信息主题化。

1　相关工作

深度学习网络在处理文本数据时，数据输入大小和输出都是先前固定的，由于文本数据与图像数据不同，文本语句长短不一，需要进行定长处理，设定序列长度。

1.1　Text-CNN 模型

Text-CNN 模型由文献[8]中提出，网络结构为“单层卷积+池化”的简易 CNN 模型，如图 1 所示。数据由词映射处理后做卷积操作得到特征向量（feature vector），再做池化操作得到文本特征表示向量。

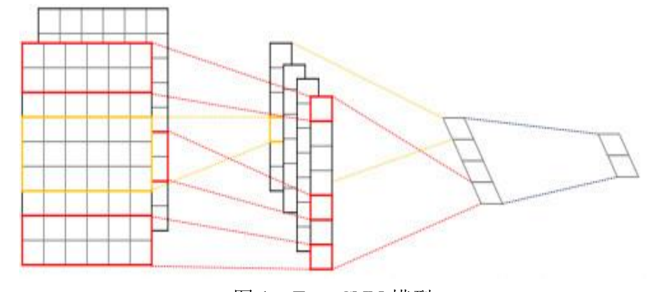


图 1　Text-CNN 模型

Fig.1　Text-CNN model

图 1 中，卷积层设计为多重过滤器宽度（同卷积核大小设定）的特征映射；池化层采用最大值池化策略，既可便于挖掘关键局部信息，又可以固定模型输出维度。CNN 虽然易于抽取局部关键特征，但存在最大问题在于卷积核大小的固定，主要影响：一是无法获取长序列信息；二是卷积核繁琐的超参调节。

1.2　LSTM 和 GRU

相比 Text-CNN，RNN 可以连贯性表达数据上下文信息。长短时记忆网络（LSTM）[18]和门控递归单元网络（GRU）[19]作为时间递归神经网络的变种，用于解决前后词语间的依赖关系，保证序列特征信息的提取，两者实际应用效果不相上下，相比 LSTM，GRU 参数量大大缩减[20]。在文本分类中的应用如图 2 所示。

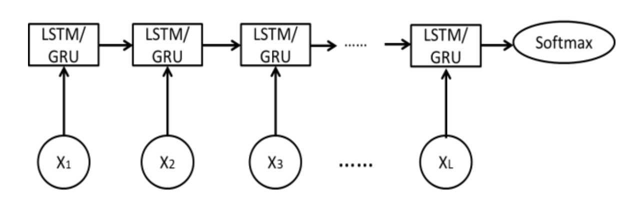


图 2　LSTM/GRU 处理文本数据

Fig.2　LSTM/GRU processing text data

图 2 中，由词向量 x1，x2….xL 输入 LSTM/GRU 模型，经处理后将词向量组合为更高一级的序列特征向量，最后实现分类。RNN 在处理数据的上下文信息，由于无法体现每一时刻输出序列信息的不同重要程度，会造成有效特征信息的丢失。所以，深度学习网络处理文本数据时，不够直观、可解释性弱。

1.3　LSTM+Attention

注意力机制（Attention mechanism）源于人类对视觉图像信息的研究，主要为了合理运用有限资源来表征整个事物本身。Attention 机制在 NLP 中的应用可以被认为是一种自动加权的方式，它可以根据每一时刻输出信息的不同重要程度，通过加权的形式对不同输出进行联系，有效提升网络对序列特征信息的学习效率。引入 LSTM[21]中应用如图 3 所示。

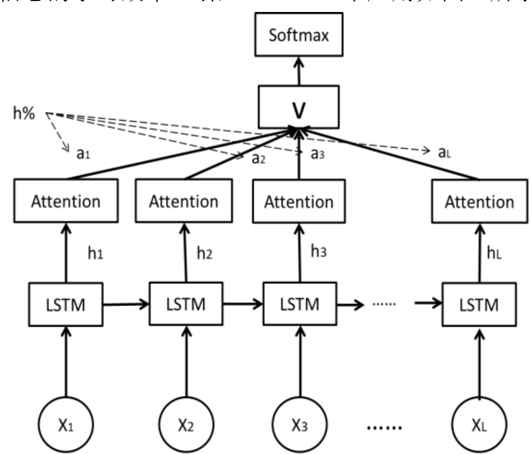


图 3　LSTM+Attention 模型

Fig.3　LSTM+Attention model

图 3 中，定义文本数据进行分词后的各个词向量为 $X_0, X_1, X_2, \ldots X_L$，然后依次输入 LSTM，对应输出定义为 h_i，下标 i 表示每一时刻输出状态，i∈[0,L]；同时在隐藏层引入 Attention 机制，计算每一时刻输出与整个特征向量的匹配得分占总体的百分率（注意力概率分布）a_i：

$$a_i=\frac{\exp(score(\bar{h},h_i))}{\sum_j \exp(score(\bar{h},h_j))} \tag{1}$$

其中：j∈[0,L]；$\bar{h}$ 可以被看做是比词向量更高一级的文本表示向量，其值可以随机初始化，并在模型训练过程中逐步更新，则第 i 时刻输出 h_i 占 $\bar{h}$ 的比重值 $score(\bar{h},h_i)$ 为

$$score(\bar{h},h_i)=w^T \mathrm{Re}LU(W\bar{h}+Uh_i+b) \tag{2}$$

其中：W、U 为权值矩阵；b 为偏置值；ReLU 为激活函数，该值越大说明了该时刻输入的文本信息注意力越大。得到每一时刻的概率分布值后，对所有时刻输出进行求和再平均，得到特征向量 V，计算公式如下：

$$V=\sum_{i=0}^{L}a_i h_i \tag{3}$$

Attention 机制的引入，使模型在处理上下文信息时，能够体现出每一时刻输出序列信息的不同权重，可以强化序列信息的有效组合。

1.4 RCNN 和 CRNN 模型

RCNN[15]和 CRNN[16]是根据 CNN、RNN 网络结构特性，通过网络层次结合方式进行有效组合的网络结构，模型保持了 CNN 处理局部信息的位置不变性和 RNN 易于序列信息建模的能力。

RCNN 模型用于解决 CNN 处理文本数据时，由于卷积核的固定，每一个词向量的上下文信息受限的问题。该模型保证了每个词向量的上下文信息，从而在提取局部信息时，期望每个词向量可以精确地表达其特征价值。CRNN 常被用于图像序列预测、文本识别等任务研究，然而在文本分类研究中，该模型通过有效提取局部特征，再对局部特征进行序列组合，减少了特征信息的丢失。然而 RNN 模型本身结构复杂，数据处理过程计算成本比较大，CNN 可以更好地体现数据信息的局部相关性，所以 CRNN 在实时性和准确率性能上要优于 RCNN，训练过程速度较快，生成模型也较小。现阶段众多任务学习中，网络结构结合模型常常被用来取代 CNN、RNN，进行模型的优化。

2 Text-CRNN+Attention

2.1 网络结构设计

针对深度学习网络在文本分类研究中展现的网络结构优势和不足，在遵循网络结构设计通用性原则下，设计了先卷积池化后循环递归的 CRNN 结构，同时引入 Attention 机制来解决模型可解释性弱问题。以循序渐进的方式，实现由局部特征提取到全局特征，再到序列信息的有效挖掘的过程。网络结构设计如图 4 所示。

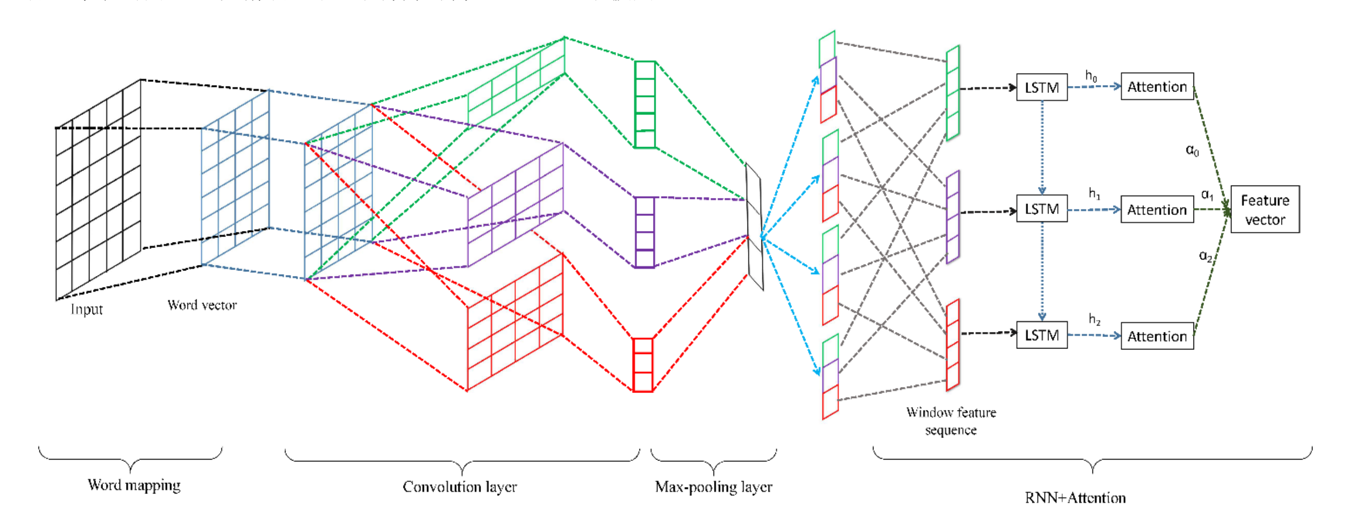


图 4 Text-CRNN+Attention 模型

Fig.4 Text-CRNN+Attention model

图 4 网络结构设计中词映射采用了 One-Hot Representation 方式；CNN 设计借鉴了文献[8]中的 Text-CNN 网络结构，采用卷积层+最大值池化层结构；RNN 模型设计采用 LSTM 结构中引入了 Attention 机制。在监督学习模式下，依据图 4 网络结构设计构建深度学习模型，则模型处理文本数据的步骤如下：

a) 将文本数据通过词映射转换为词向量，作为网络输入；

b) 利用 CNN 处理局部特征的位置不变性，提取高效局部特征信息，得到特征向量（feature vector）作为 RNN 网络输入；

c) 将 Attention 机制引入 RNN 网络，根据每一时刻输出不同重要程度进行自动加权，强化序列特征建模；

d) Softmax 分类器来输出网络预测每一类别的概率。

2.2 文本数据处理过程

通过深度学习框架构建序贯模型（sequential）， 在模型进行训练和执行预测时，Text-CRNN+Attention 模型在处理文本数据的具体实现过程为：

a）词映射。

将文本数据进行词映射（将词语向量化并降维处理）处理，假定文本语句中有 N 个词，分布式表示为 N 维向量 Z_N：

$$Z_N=[0,0,1,0,\cdots,0,\cdots 0] \tag{4}$$

定义维度为 m、第 i 词语对应词向量为 $a_i \epsilon R m$，则长度为 j 的语句中[i,j]词语的拼接序列词向量为

$$a_{i:j}=a_i \oplus a_{i+1} \oplus a_{i+2} \oplus \cdots \oplus a_j \tag{5}$$

b）词向量经 CNN 网络提取高效局部特征。

完成词映射后，将词向量输入 CNN 模型进行处理，卷积层中卷积核对应词个数记为 n；卷积核对应的权值矩阵为 $h \epsilon R_{nm}$；令偏置值 $b \epsilon R_m$；对比实验中所用为 ReLU 激活函数假定为 f，则卷积核在词序列中第 i 位置上的输出为

$$T_i=f(h \cdot a_{i:i+n-1}+b) \tag{6}$$

池化层采用了最大值池化（max-pooling）：

$$\tilde{T}_i=\max\{T_i\} \tag{7}$$

文本特征向量池化操作与图像数据池化不同，处理过程是将卷积后的输出向量经池化后直接得到一个数值，得到全局特征向量 Feature vector：

$$T=[\tilde{T}_1,\tilde{T}_2,\tilde{T}_3,\cdots,\tilde{T}_{j-n+1}] \tag{8}$$

c)将 Attention 机制引入 RNN 网络，强化序列特征建模。

经过 CNN 模型得到 Feature vector，以 CNN 输出为 LSTM 输入，定义 Feature vector 输入 LSTM 后挖掘的特征序列为 T_m，每一时刻输入特征向量为 $\widetilde{T}_{m-1}$（对应为 $\widetilde{T}_{j-n+1}$）。

LSTM 网络处理文本信息时，通过四个彼此交互的“门”单元来控制每一输入对特征序列的影响变化。令 l_t 为遗忘门（forget gate layer），f 为激活函数，将特征序列 T_{m-1} 与当前输入特征向量的线性组合为

$$l_t = f(W_l^T \times T_{m-1} + U_l^T \times \widetilde{T}_m + b_l) \tag{9}$$

其中：W、U 为连接上层隐层与当前隐层的权重矩阵；b 为偏置参数；同理，输入门（input gate layer）i_t、候选门（candidate layer）c'_t、输出门（output gate layer）o_t 运行策略分别表示为

$$i_t = f(W_i^T \times \widetilde{T}_{m-1} + U_i^T \times T_m + b_i) \tag{10}$$

$$c_t^{'} = f(W_c^T \times \widetilde{T}_{m-1} + U_c^T \times T_m + b_c) \tag{11}$$

$$o_t = f(W_o^T \times T_{m-1} + U_o^T \times a_m + b_o) \tag{12}$$

线性组合后的记忆特征 c_t 由遗忘门过滤后的部分记忆和 $l_t \times c_{t-1}$ 和新增加的特征信息 $i_t \times c'_t$（新增比例由输入门控制）构成：

$$c_t = l_t \times c_{t-1} + i_t \times f(W_c^T \times \widetilde{T}_{m-1} + U_c^T \times T_m + b_c) \tag{13}$$

则每一时刻输出门输出并传递给下一阶段的特征序列信息为

$$\begin{aligned}\widetilde{T}_m &= o_t \times f(c_t) \\ &= o_t \times f(l_t \times c_{t-1} + i_t \times c_t^{'})\end{aligned} \tag{14}$$

则输出的序列特征信息组合为

$$Z = \{\widetilde{T}_1, \widetilde{T}_2, \cdots, \widetilde{T}_m\} \tag{15}$$

在 LSTM 模型中引入 Attention 机制，对应于式（1）。a_i 计算公式为

$$a_i = \frac{\exp(score(\overline{T}, \widetilde{T}_i))}{\sum_j \exp(\overline{T}, \widetilde{T}_i)} \tag{16}$$

其中：j∊[0,L]；$\overline{T}$ 可以被看做是比输入特征向量更高一级的文本表示向量，其值也是随机初始化；所以 $score(\overline{T}, \widetilde{T}_i)$ 为

$$score(\overline{T}, \widetilde{T}_i) = w^T \mathrm{ReLU}(W\overline{T} + U\widetilde{T}_i + b) \tag{17}$$

得到每一时刻的概率分布值后，对每一时刻输出进行求和再平均后，得到特征向量 V：

$$V = \sum_{i=0}^{L} a_i \widetilde{T}_i \tag{18}$$

d）由 Softmax 分类器来预测输出。

经过网络处理得到特征向量后，由 Softmax 分类器预测文本数据分类概率：

$$y = Soft\max(W_V V + b_V) \tag{19}$$

该输出为网络预测的每一类别概率，选择具有最高概率的输出。

3　实验结果及分析

3.1　数据集

在监督学习模式下，对比实验数据集采用清华大学自然语言处理实验室推出的 THUCT 中文文本数据库中的部分子数据集。该数据集文本数量大、类别多；包含财经、房产、体育、教育、时尚、科技、娱乐、游戏、时政、家居共计 10 类。数据集被划分为训练、验证、测试集。数据集划分基本信息如表 1 所示。

表 1　数据集划分信息

Table 1　Data set partitioning information

类型	类别	训练集	验证集	测试集
数量	10	5000×10	500×10	1000×10

3.2　实验环境及参数设定

为了合理有效验证模型的性能指标，对比实验环境设置：操作系统为 Ubuntu16.04，内存 3.7 GB，处理器为 Intel®Core™i5-7200U CPU@2.50 GHz×4，在 Tensorflow 框架下使用 Python 语言。

模型训练、执行预测时，文本序列长度统一设置为 600；学习率的设置直接影响着网络模型的优化效果，其设置不能过大也不能过小，对比实验采用了学习率的指数衰减；卷积层和池化层相关参数借鉴文献[8]，并不断调优后，设定卷积核尺寸（长度）为 4；其宽度和词向量一致为 64；卷积核数目设置为 256；最大的池化层设置为 4；LSTM 层、GRU 层神经元个数设置为 128；训练批次周期为 20，每迭代 100 轮输出一次结果。

3.3　模型对比实验分析

为了保证测试数据评估效果与在真实场景下模型对未知数据预判的结果相近，模型训练时，数据集划分了训练集、验证集。图 5 和 6 分别为七种模型训练集、验证集准确率（accuracy）随迭代次数（iter）变化曲线。最后对模型进行测实验证，模型训练及测试时间、测试准确率如表 2 所示。表 3 为每一类别在 Text-CRNN+Attention 模型及六种对比模型中的测试识别率对比结果。

在图 5、6 中可以直观地看出，训练准确率变化曲线和验证准确率变化曲线在迭代 200 轮时，相比其他六种模型，Text-CRNN+Attention 模型收敛速度较快，达到一个相对较高的准确率，并在 200 轮之后曲线逐步趋于平滑状，LSTM 和 GRU 模型相对波动幅度最大，相比 Text-CNN、RCNN、CRNN 模型，Text-CNN+RNN 模型平滑程度较为明显，训练准确率和预测准确率基本持平。由此表明，Text-RNN+Attention 模型网络学习能力和模型泛化能力明显优于其他六种模型。

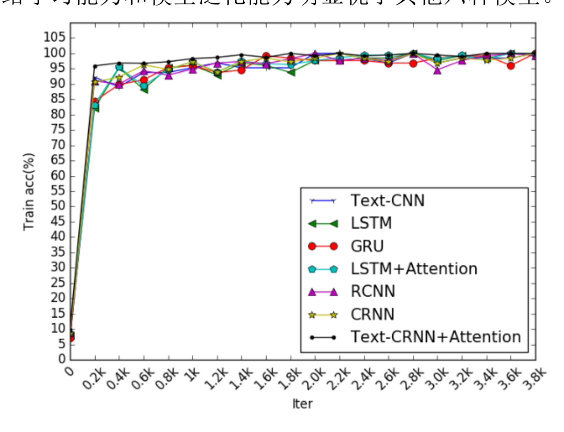


图 5　训练集准确率变化曲线

Fig.5　Accuracy change curve of training set

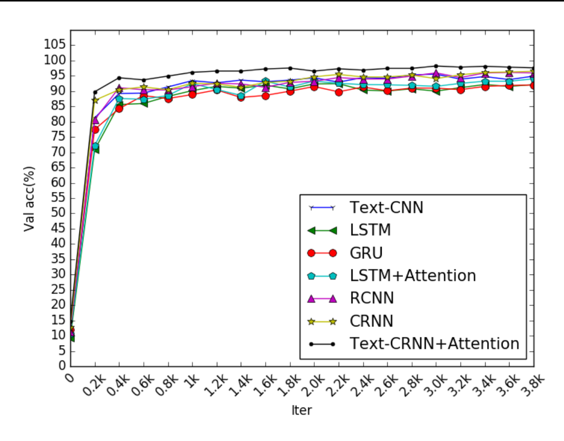


图 6　验证集预测准确率变化曲线

Fig. 6　Verification set prediction accuracy change curve

表 2 数据显示，CNN 模型以自身结构的先验条件，在实时性上占据显著优势，准确率也相对较高；RNN（LSTM/GRU）结构自身较为复杂，模型训练和测试耗时较长，引入 Attention 机制后，文本数据处理计算成本较大；RCNN、CRNN 模型有效提升了准确率，验证了 CNN、RNN 以网络层次结合的变体结构模型的可靠性。相比六种对比模型，虽然 Text-CRNN+Attention 模型由于网络结构复杂性增加，引入 Attention 机制强化网络学习能力的同时模型计算量指数级增加，所以在实时性没有取得优势，但准确率相对提升了 2~3 个百分点，证实了模型的优越性。表 3 数据显示，相比六种对比模型，Text-CRNN+Attention 模型相对提升了每一类别的准确率。整体效果而言，七种模型在房产和游戏类的信息分类中效果最佳，教育类信息分类总体效果不佳，Text-CRNN+Attention 模型略占优势。

表 2　模型训练及测试对比结果

Table 2　Model training and test comparison results

模型	训练时间/s	测试时间/s	测试准确率/%
Text-CNN	4720	51	96.53
LSTM	14963	216	95.64
GRU	15086	174	95.04
LSTM+Attention	16561	235	96.47
RCNN	18056	265	97.12
CRNN	12439	127	97.32
Text-CRNN+Attention	13858	156	98.41

表 3　每一类别识别率对比结果

Table 3　Comparison rate of each category recognition rate

类别准确率/%	财经	房产	体育	教育	时尚	科技	娱乐	游戏	时政	家居
Text-CNN	93	100	99	88	94	94	96	99	96	98
LSTM	98	99	99	89	95	94	97	97	92	98
GRU	95	100	98	85	92	92	97	98	93	97
LSTM+Attention	98	100	99	90	96	94	97	98	95	98
RCNN	96	100	99	88	94	95	97	98	93	98
CRNN	96	100	99	89	95	95	97	99	96	98
Text-CRNN+Attention	99	100	99	92	96	96	98	99	97	99

4　结束语

在多类别中文文本分类任务研究中，将 Attention 机制引入 CRNN 网络，本文提出了一种 Text-CRNN+Attention 模型。实验结果表明，该模型在处理文本数据时，既保持了 CNN 处理局部特征的位置不变性，又起到增强 RNN 挖掘序列信息的能力，具有较强的网络学习能力和模型泛化能力；同时验证了在数据具有空间和时间上的连续性时，由于 Attention 机制计算过程是可微的，所以可适用于不同网络结构中，可以不断扩展应用领域；虽然该模型分类效果有一定的提升，但 CRNN 网络结构设计本身具有一定的复杂性，同时 Attention 机制的引入需要耗费一定的计算量，所以在实时性上没有取得很好地效果。所以接下来有待于深入研究的问题在于：a）针对 Attention 机制计算量随输入输出序列增加呈指数级增长问题进行优化模型，改善实时性；b）如何选择性地关注关键输出序列信息，而不是每一时刻都进行。

参考文献：

[1] Aliwy A H, Ameer E H A. Comparative study of five text classification algorithms with their improvements [J]. International Journal of Applied Engineering Research, 2017, 12 (14): 4309-4319.

[2] Rajvanshi N, Chowdhary K R, Rajvanshi N, *et al.* Comparison of SVM and naive Bayes text classification algorithms using WEKA [J]. International Journal of Engineering & Technical Research, 2017, 6 (9): 141-143.

[3] Goudjil M, Koudil M, Bedda M, *et al.* Anovel active learning method using SVM for text classification [J]. International Journal of Automation and Compution, 2016: 1-9.

[4] Li Dawei, Hu Xiaojian, Jin Chengjie, *et al.* Learning to detect traffic incidents from data based on tree augmented naive Bayesian classifiers [J]. Discrete Dynamics in Nature & Society, 2017 (1): 1-9.

[5] Joulin A, Grave E, Bojanowski P, *et al.* Bag of tricks for efficient text classification [J]. arXiv preprint arXiv, 2016: 1607. 01759.

[6] Mikolov T, Sutskever I, Chen Kai, *et al.* Distributed representations of words and phrases and their compositionality [J]. Advances in Neural Information Processing Systems, 2013, 26: 3111-3119.

[7] Mikolov T, Chen Kai, Corrado G, *et al.* Efficient estimation of word representations in vector space [J]. arXiv preprint arXiv, 2013: 1301. 3781.

[8] Kim Y. Convolutional neural networks for sentence classification [J]. arXiv preprint arXiv, 2014: 1408. 5882.

[9] Kartsaklis D, Kalchbrenner N, Sadrzadeh M. Resolving lexical ambiguity in tensor regression models of meaning [J]. arXiv preprint arXiv, 2014: 1408. 6181.

[10] Sutskever I, Martens J, Hinton G E. Generating text with recurrent neural networks [C]// Proc of International Conference on Machine Learning . 2011: 1017-1024.

[11] Liu Pengfei, Qiu Xipeng, Huang Xuanjing. Recurrent neural network for text classification with multi-task learning [J]. arXiv preprint arXiv, 2016: 1605. 05101.

[12] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate [J]. arXiv preprint arXiv, 2014: 1409. 0473.

[13] Yin Wenpeng, Schyutze H, Xiang Bing, *et al.* Abcnn: Attention-based convolutional neural network for modeling sentence pairs [J]. Transactions of the Association for Computational Linguistics, 2016, 4: 259-272.

[14] 王伟，孙玉霞，齐庆杰，等. 基于 BiGRU-Attention 神经网络的文本情感分类模型 [J/OL]. 计算机应用研究，2018. (2019-02-20) [2019-02-20]. http://kns. cnki. net/kcms/detail/51. 1196. TP. 20181011. 1246. 010. html. (Wang Wei, Sun Yuxia, Qi Qingjie, *et al.* Text sentiment classification model based on BiGRU-Attention neural network [J/OL]. Application Research of Computers, 2018.

(2019-02-20) [2019-02-20]. http://kns. cnki. net/kcms/detail/51. 1196. TP. 20181011. 1246. 010. html.)

[15] Lai Siwei, Xu Liheng, Liu Kang, *et al.* Recurrent convolutional neural networks for text classification [C]// Proc of the 29th AAAI Conference on Artificial Intelligence. 2015: 579-585.

[16] Shi Baoguang, Xiang Bai, Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2015, 39 (11): 2298-2304.

[17] Szegedy C, Liu Wei, Jia Yangqing, *et al.* Going deeper with convolutions [C]// Proc of the 28th IEEE Conference on Computer Vision and Pattern Recognition. 2015: 1-9.

[18] Hochreiter S, Schmidhuber J. Long short-term memory [J]. Neural Computation. 1997, 9 (8): 1735-1780.

[19] Chung J, Gulcehre C, Cho K, *et al.* Supplementary material: Gated feedback recurrent neural networks [C]// Proc of International Conference on Machine Learning. 2015: 2067-2075.

[20] Irie K, Tyuske Z, Alkhouli T, *et al.* LSTM, GRU, highway and a bit of attention: an empirical overview for language modeling in speech recognition [C]// Proc of the 17th Annual Conference of the International Speech Communication Association. 2016: 3519-3523.

[21] 马建红，王瑞杨，姚爽，等. 基于深度学习的专利分类方法 [J]. 计算机工程, 2018, 44 (10): 209-214. (Ma Jianhong, Wang Ruiyang, Yao Shuang, *et al.* Patent classification method based on depth learning [J]. Computer Engineering, 2018, 44 (10): 209-214.)